

Amendments to the Claims

This listing of claims will replace all prior versions and listings of claims in the application.

Claim 1 (previously presented): A cache controller for use with a processor, comprising:

a plurality of mappers for receiving instructions of an instruction set, each mapper for mapping an instruction of said instruction set to a predetermined instruction width format (PIWF) configuration, wherein said plurality of mappers include

at least one first mapper for receiving instructions from a fill buffer, and

at least one second mapper for receiving instructions from an instruction cache; and

a multiplexor for receiving said PIWF configurations from said plurality of mappers and selecting, in response to a selector signal, a desired one of said PIWF configurations for decoding and execution by the processor.

Claim 2 (original): The cache controller of claim 1, further comprising:

a tag comparator for generating said selector signal.

Claim 3 (previously presented): The cache controller of claim 2, wherein said tag comparator comprises:

means for comparing, for each instruction provided to one of said plurality of mappers, a tag associated with an instruction of said instruction set to a desired tag and generating said selector signal to cause said multiplexor to select said desired one of said PIWF configurations.

Claim 4 (canceled)

Claim 5 (previously presented): In a cache controller for use with a processor, a method for mapping an instruction set to a predetermined instruction width format (PIWF) configuration, comprising:

(a) reading instructions of said instruction set from an instruction cache and a fill buffer into a plurality of mappers, wherein at least one of said instructions is read from said instruction cache and at least one of said instructions is read from said fill buffer, each instruction of said instruction set being read into a corresponding one of said plurality of mappers in preparation for mapping;

(b) mapping each instruction of said instruction set to a corresponding PIWF configuration; and

(c) selecting a desired one of said PIWF configurations for decoding and execution by the processor.

Claim 6 (currently amended): The method of claim 5, further comprising ~~the step of~~:

(d) comparing, for each instruction provided to one of said plurality of mappers, a tag associated with an instruction of said instruction set to a desired tag, wherein said desired one of said PIWF configurations is selected based on said comparison.

Claim 7 (canceled)

Claim 8 (previously presented): A processor comprising:

an execution unit;

a decoder;

a cache for storing instructions; and

a cache controller for retrieving said instructions from said cache and providing said instructions to said decoder, said cache controller comprising:

a plurality of mappers for mapping a plurality of instructions of an instruction set to predetermined instruction width format (PIWF) configurations, said plurality of mappers including at least one first mapper for receiving instructions from a fill buffer, and at least one second mapper for receiving instructions from said instruction cache,

a multiplexor for selecting, in response to a selector signal, one of said PIWF configurations for decoding by said decoder and execution by said execution unit, and

means for comparing, for each instruction provided to said multiplexor, a tag associated with an instruction of said instruction set to a desired tag and generating said selector signal to cause said multiplexor to select said desired one of said PIWF configurations,

whereby said processor performs instruction mapping substantially in parallel with tag comparison to improve processor performance.

Claim 9 (canceled)

Claim 10 (currently amended): A tangible computer readable medium comprising a microprocessor core embodied in software, the microprocessor core including a cache controller comprising:

a plurality of mappers for receiving instructions of an instruction set, each mapper for mapping an instruction of said instruction set to a predetermined instruction width format (PIWF) configuration, wherein said plurality of mappers include

at least one first mapper for receiving instructions from a fill buffer, and

at least one second mapper for receiving instructions from an instruction cache; and

a multiplexor for receiving said PIWF configurations from said plurality of mappers and selecting, in response to a selector signal, a desired one of said PIWF configurations for decoding and execution by the processor.

Claim 11 (currently amended): The tangible computer readable medium of claim 10, wherein said cache controller further comprises:

a tag comparator, configured to compare, for each instruction provided to one of said plurality of mappers, a tag associated with an instruction of said instruction set to a desired tag and ~~generating~~ to generate said selector signal to cause said multiplexor to select said desired one of said PIWF configurations.

Claim 12 (new): The tangible computer readable medium of claim 10, wherein said microprocessor core is embodied in hardware description language software.

Claim 13 (new): The tangible computer readable medium of claim 12, wherein said microprocessor core is embodied in Verilog hardware description language software.

Claim 14 (new): The tangible computer readable medium of claim 12, wherein said microprocessor core is embodied in VHDL hardware description language software.

Claim 15 (new): A method for providing a microprocessor core including a cache controller, the method transmitting the microprocessor core over a communications network, the method comprising:

providing computer-readable program code describing the microprocessor core including the cache controller, wherein the cache controller comprises

a plurality of mappers for receiving instructions of an instruction set, each mapper for mapping an instruction of the instruction set to a predetermined instruction width format (PIWF) configuration, wherein the plurality of mappers include

at least one first mapper for receiving instructions from a fill buffer,
and

at least one second mapper for receiving instructions from an
instruction cache, and

a multiplexor for receiving the PIWF configurations from the plurality of
mappers and selecting, in response to a selector signal, a desired one of the PIWF
configurations for decoding and execution by the microprocessor core; and

transmitting the computer-readable program code over the communications network.

Claim 16 (new): The method of claim 15, wherein the cache controller further comprises:

a tag comparator, configured to compare, for each instruction provided to one of the
plurality of mappers, a tag associated with an instruction of the instruction set to a desired tag
and to generate the selector signal to cause the multiplexor to select the desired one of the
PIWF configurations.

Claim 17 (new): The method of claim 15, wherein the computer-readable program code is
hardware description language code.

Claim 18 (new): The method of claim 17, wherein the computer-readable program code is
Verilog hardware description language code.

Claim 19 (new): The method of claim 17, wherein the computer-readable program code is
VHDL hardware description language code.

Claim 20 (new): The method of claim 15, wherein the transmitting step comprises transmitting the computer-readable program code over the Internet.